CLAIMS

What is claimed is:

1   1.   A method including:

2

3   maintaining a state machine to provide a multi-bit output, each bit of

4   the multi-bit output indicating a respective status of an associated

5   thread of multiple threads being executed with a multithreaded

6   processor;

7

8   detecting a change of status for a first thread within the

9   multithreaded processor; and

10

11   configuring a functional unit within the multithreaded processor in

12   accordance with the multi-bit output of the state machine.

1   2.   The method of claim 1 wherein each bit of the multi-bit output

2   indicates the status of the associated thread as being active or inactive.

1   3.   The method of claim 2 wherein the configuring of the functional unit

2   comprises partitioning the functional unit to service both the first thread and

3   a second thread within the multithreaded processor when the change of

4   status for the first thread comprises a transition from an inactive state to an

5   active state.

1   4.   The method of claim 2 wherein the configuring of the functional unit

2   comprises un-partitioning the functional unit to service a second thread, but

3   not the first thread, within the multithreaded processor when the change of

4   the status of the first thread comprises a transition from an active state to an

5   inactive state.

1   5.   The method of claim 1 wherein the detecting of the change in the

2   status of the first thread comprises detecting the occurrence of an event for

3   the first thread.

1   6.   The method of claim 5 including asserting a first signal responsive to

2   the occurrence of the event for the first thread, and evaluating the state

3   machine during the assertion of the first signal.

1   7.   The method of claim 6 wherein the functional unit within the

2   multithreaded processor is configured, in accordance with the multi-bit

3   output of the state machine, on the de-assertion of the first signal.

1   8.   The method of claim 1 wherein the detecting of the change in the

2   status of the first thread comprises detecting the occurrence of a sleep event

3   for the first thread that transitions the first thread from an active state to a

4   sleep state.

1   9.   The method of claim 8 including, responsive to the detection of the
2   occurrence of the sleep event, setting an inhibit register to inhibit an event
3   that is not a break event for the sleep state of the first thread.

1   10.   The method of claim 1 wherein the configuring of the functional unit
2   within the multithreaded processor comprises saving and deallocating state
3   within the multithreaded processor for the first thread.

1   11.   The method of claim 10 wherein the saving and deallocating of the
2   state within the multithreaded processor for the first thread comprises
3   recording the state for the first thread within a memory resource.

1   12.   The method of claim 1 wherein the configuring of the functional unit
2   within the multithreaded processor comprises making registers, within a
3   register file of the multithreaded processor, available to a second thread
4   within the multithreaded processor.

1   13.   The method of claim 1 wherein the functional unit comprises any one
2   of the group of functional units including a memory order buffer, a store
3   buffer, a translation lookaside buffer, a reorder buffer, a register alias table,
4   and a free list manager.

-72-

1　14.　The method of claim 1 wherein the configuring of the functional unit

2　includes inserting a fence instruction into an instruction stream for the first

3　thread at a location proximate a front-end of the multithreaded processor,

4　the fence instruction defining an event boundary within the instruction

5　stream that assumes all memory accesses have drained from the processor..

1　15.　The method of claim 1 wherein the configuring of the functional unit

2　includes restoring state within the multithreaded processor.

1　16.　The method of claim 1 wherein the detecting of the change in the

2　status of the first thread comprises detecting the occurrence of a break event

3　for the first thread that transitions the first thread from a sleep state to an

4　active state.

1　17.　The method of claim 16 including detecting a third event for the first

2　thread that does not constitute a break event, and logging the third event

3　within a pending register associated with the first thread.

1　18.　Apparatus comprising:

2

3　　　a state machine to provide a multi-bit output, each bit of the multi-

4　　　output indicating a respective status of an associated thread of

5   multiple threads being executed within a multithreaded processor,

6   and to detect a change of status for a first thread within the

7   multithreaded processor; and

8

9   configuration logic to configure a functional unit within the

10   multithreaded processor in accordance with the multi-bit output of

11   the state machine.

1   19.   The apparatus of claim 18 wherein each bit of the multi-bit output

2   indicates the status of the associated thread as being active or inactive.

1   20.   The apparatus of claim 19 wherein the configuration logic partitions

2   the functional unit to service both the first thread and a second thread within

3   the multithreaded processor when the change of status for the first thread

4   comprises a transition from an inactive state to an active state and the second

5   thread is in an active state.

1   21.   The apparatus of claim 19 wherein the configuration logic un-

2   partitions the functional unit to service a second thread, but not the first

3   thread, within the multithreaded processor when the change of the status of

4   the first thread comprises a transition from an active state to an inactive state

5   and the second thread is in an active state.

1   22.   The apparatus of claim 18 wherein the state machine detects the
2   change in the status of the first thread by detecting the occurrence of an
3   event for the first thread.

1   23.   The apparatus of claim 22 including an event detector that asserts a
2   clearing signal responsive to the occurrence of the event for the first thread,
3   and wherein the state machine is evaluated during the assertion of the first
4   signal.

1   24.   The apparatus of claim 23 wherein the configuration logic configures
2   the functional unit within the multithreaded processor in accordance with
3   the multi-bit output of the state machine on the de-assertion of the clearing
4   signal.

1   25.   The apparatus of claim 18 wherein the state machine, to detect the
2   change in the status of the first thread, detects the occurrence of a sleep event
3   for the first thread that transitions the first thread from an active state to a
4   sleep state.

1   26.   The apparatus of claim 25 including a microcode sequencer that,
2   responsive to the detection of the occurrence of the sleep event, issues a
3   microinstruction to set an inhibit register to inhibit an event that is not a
4   break event for the sleep state of the first thread.

1   27.   The apparatus of claim 18 wherein the configuration logic saves,

2   deallocates and restores state within an associated functional unit for the

3   first thread.

1   28.   The apparatus of claim 27 wherein the configuration logic associated

2   with the functional unit records state information for the first thread within

3   a memory resource to save and deallocate state, and restores state

4   information for the first thread to functional unit from the memory resource

5   to restore state.

1   29.   The apparatus of claim 27 wherein the configuration logic associated

2   with the functional unit makes registers, within a register file of the

3   multithreaded processor, allocated to the first thread available to a second

4   thread within the multithreaded processor if the first thread exits and makes

5   registers, within the register file of the multithreaded processor, allocated to

6   the second thread available to the first thread within the multithreaded

7   processor if the second thread exits.

1   30.   The apparatus of claim 18 wherein the functional unit comprises any

2   one of the group of functional units including a memory order buffer, a store

3   buffer, a translation lookaside buffer, a reorder buffer, a register alias table,

4   and a free list manager.

1 31. The apparatus of claim 18 including a microcode sequencer that

2 introduces a fence instruction into an instruction stream for the first thread

3 at a location proximate a front-end of the multithreaded processor, the fence

4 instruction defining an event boundary within the instruction stream to

5 ensure that all memory accesses drain from the processor..

1 32. The apparatus of claim 18 wherein the configuring of the functional

2 unit includes restoring state within the multithreaded processor.

1 33. The apparatus of claim 23 wherein the event detector detects the

2 change in the status of the first thread by detecting the occurrence of a break

3 event for the first thread that transitions the first thread from a sleep state to

4 an active state

1 34. The apparatus of claim 23 wherein the event detector detects a third

2   event for the first thread that does not constitute a break event, and

3   logs the third event within a pending register associated with the first

4   thread.

1 35. Apparatus comprising:

2

3   first means for providing a multi-bit output, each bit of the multi-

4   output indicating a respective status of an associated thread of

5   multiple threads being executed within a multithreaded processor,

6   and to detect a change of status for a first thread within the

7   multithreaded processor; and

8

9   second means for configuring a functional unit within the

10   multithreaded processor in accordance with the multi-bit output of

11   the state machine.

1   36.   A machine-readable medium including a sequence of instructions

2   that, when executed by a machine, cause the machine to:

3

4   maintain a state machine to provide a multi-bit output, each bit of the

5   multi-bit output indicating a respective status of an associated thread

6   of multiple threads being executed with a multithreaded processor;

7

8   detect a change of status for a first thread within the multithreaded

9   processor; and

10

11   configure a functional unit within the multithreaded processor in

12   accordance with the multi-bit output of the state machine.